

# Dimensionality Reduction with Neuro-Fuzzy Discriminant Analysis

Rami N. Khushaba, Adel Al-Jumaily, and Ahmed Al-Ani

**Abstract**—One of the most important tasks in any pattern recognition system is to find an informative, yet small, subset of features with enhanced discriminatory power. In this paper, a new neuro-fuzzy discriminant analysis based feature projection technique is presented based on a two stages hybrid of Neural Networks, optimized with Differential Evolution (DE), and a proposed Fuzzy Linear Discriminant Analysis (FLDA) technique. Although dimensionality reduction via FLDA can present a set of well clustered features in the reduced space, but like any version of the existing DA's it assumes that the original data set is linearly separable, which is not the case with many real world problems. In order to overcome this problem, the first stage of the proposed technique maps the initially extracted features in a nonlinear manner into a new domain, with larger dimensionality, in which the features are linearly separable. FLDA acts then on these linearly separable features to further reduce the dimensionality. The proposed combination, referred to as NFDA, is validated on a prosthetic device control problem with Electroencephalogram (EEG) datasets collected from 5 subjects achieving a maximum testing accuracy of 85.7% for a three classes of EEG based imaginations of movements.

**Keywords**—Feature Projection, Linear Discriminant Analysis, Prostheses Control.

## I. INTRODUCTION

**T**ECHNIQUES that can introduce low-dimensional feature representation with enhanced discriminatory power are of paramount importance, because of the so called curse of dimensionality [1]. Various methods have been proposed for dimensionality reduction and feature extraction, such as Principal Component Analysis (PCA), Independent Component Analysis (ICA), and Linear Discriminant Analysis (LDA). LDA, unlike other methods, is particularly suitable for solving classification problems. It aims to maximize the ratio of the determinant of the between-class scatter matrix to the determinant of the within-class scatter matrix of the projected samples. However, there are many problems with the so called classical LDA. Firstly, in certain situations the number of data points is smaller than the dimension of the data space, this in turn causes all scatter matrices to be singular and we have the under-sampled or singularity problem. Classical LDA requires the scatter matrices to be non-singular and fails when the scatter matrices are singular. Secondly, classical LDA pays no attention to the decorrelation of the data. In many applications it is always desirable that the features would be of minimum correlation to reduce the redundancy in the extracted information. Another limitation with LDA is that it treats all the data points equivalently where as in the real world

problems each sample may belong to each of the different classes to a certain degree.

In order to address the third limitation, there were only few attempts in the literature that proposed a fuzzy version of LDA, termed in this paper as FLDA. Examples of FLDA include the work presented by both Watada et al [2] and Wua and Zhoua [3]. Although successful in many applications and being an enhanced version of the classical LDA, FLDA lacks the capacity to capture a nonlinearly clustered structure in the data because of its linear nature. Motivated by extracting nonlinear features, there were only countable attempts to solve this problem by employing kernel based approaches [4], [5], [6]. Due to the computational complexity associated with the kernel based approaches, especially for very large datasets, then it would be tempting to search for alternative methods to perform the nonlinear mapping task.

In this paper, a two layer projection technique is presented. In the first layer a feed forward neural network layer is utilized as a nonlinear mapping stage for which the parameters are optimized with Differential Evolution (DE) [7]. The goal behind using this layer is to nonlinearly map the input space to a high-dimensional feature space where different classes of objects are supposed to be linearly separable. This will prepare the scene for the second stage for further reducing the dimensionality by utilizing a new version of FLDA that can identify outliers and reduce their effects on the formed cluster structure.

This paper is organized as follows: Section II introduces the proposed projection technique and the DE optimization. The experiments and practical results are given in section III. Finally a conclusion is given in Section IV.

## II. NEURO-FUZZY DISCRIMINANT ANALYSIS BASED FEATURE PROJECTION

An artificial neural network (ANN) model is an information processing paradigm inspired by the way the biological nervous system process information. It consists of many nonlinear computational elements operating in parallel and arranged in patterns reminiscent of biological neural networks. These computational elements, known as the nodes or the neurons, are connected via weights that are typically adapted during the use to improve the performance. The ANNs have long been utilized in a great variety of tasks. However, at present, their main practical applications have been for classification tasks.

Earlier studies on the relation between discriminant analysis and the Multilayer Perceptrons (MLP) used for classifications date back long time ago. Several studies were made to illustrate why nonlinear adaptive feed-forward layered networks

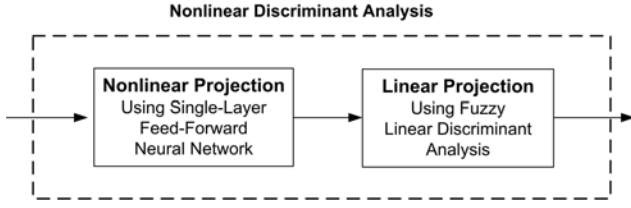


Fig. 1. Block Diagram of the proposed projection technique

with linear output units can perform well for pattern classification [8], [9]. These studies proved that within MLP, each layer of weights can be thought of as performing projections that try to separate as best as possible the different classes, so they can be linearly separable by the cells in the last layer. All of these studies suggest that the MLP actually consist of two projections: A Non-linear projection from input-to-hidden and from each hidden-to-hidden layer and a second projection being linear from the final hidden-to-output layer.

Several studies followed, but the main trend was decomposed into two parts. The first focused on enhancing the functionality of multilayer feed-forward neural networks performing the nonlinear discriminant analysis [10], [11]. The second trend, as mentioned earlier, focused on Fisher's discriminant analysis itself as a statistical technique and mixing this technique with kernel function to perform the nonlinear mapping [4], [5], [6]. Although many of these studies does actually perform well as a nonlinear discriminant analysis tool, but up to the authors' knowledge there were no studies that combined neural networks with the statistical discriminant analysis to form a dimensionality reduction tool. Thus the main focus of this paper is to combine these two techniques and compare the performance of the proposed nonlinear method with other techniques.

The basic structure proposed in this paper is shown in Figure. 1 sharing similar architecture with the multilayer perceptrons (MLPs). Only one hidden layer is utilized. Also since the final layer implements a linear mapping, the final layer was replaced with our new version of FLDA. The main reason behind this is to reduce the computational cost associated with the optimization process, since the connection weight values are evolved using the Differential Evolution (DE) optimization technique. Thus, the weights of the hidden layer are optimized according to the given problem. Then FLDA acts upon the output of this hidden layer to perform the rest of the projection task. In the next section, the DE based optimization is introduced.

#### A. Differential Evolution based Weights Optimization

Differential evolution is a simple optimization technique having parallel, direct search, easy to use, good convergence, and fast implementation properties [7]. The crucial idea behind DE is a new scheme for generating trial parameter vectors by adding the weighted difference vector between two population members  $\mathbf{x}_{r1}$  and  $\mathbf{x}_{r2}$  to a third member  $\mathbf{x}_{r0}$ . The following equation shows how to combine three different, randomly chosen vectors to create a mutant vector,  $\mathbf{v}_{i,g}$  from the current generation  $g$ :

$$\mathbf{v}_{i,g} = \mathbf{x}_{r0,g} + F \times (\mathbf{x}_{r1,g} - \mathbf{x}_{r2,g}) \quad (1)$$

where  $F \in (0, 1)$  is a scale factor that controls the rate at which the population evolves. The index  $g$  indicates the generation to which a vector belongs. In addition, each vector is assigned a population index,  $i$ , which runs from 0 to  $N_p - 1$ . Parameters within vectors are indexed with  $j$ , which runs from 0 to  $D - 1$ .

Extracting both distance and direction information from the population to generate random deviations results in an adaptive scheme that has excellent convergence properties. In addition, DE also employs uniform crossover, also known as discrete recombination, in order to build trial vectors out of parameter values that have been copied from two different vectors. In particular, DE crosses each vector with a mutant vector, as given below:

$$\mathbf{u}_{i,g} = \mathbf{u}_{j,i,g} = \begin{cases} \mathbf{v}_{j,i,g} & \text{if } \text{rand}(0,1) \leq C_r \text{ or} \\ \mathbf{x}_{j,i,g} & \text{Otherwise} \end{cases} \quad (2)$$

where  $\mathbf{u}_{j,i,g}$  is the  $i$ th trial vector along  $j$ th dimension from the current population  $g$ . The crossover probability  $C_r \in [0, 1]$  is a user defined value that controls the fraction of parameter values that are copied from the mutant. If the newly generated vector results in a lower objective function value (higher fitness) than the predetermined population member, then the resulting vector replaces the vector with which it was compared [12].

Each member of the population hold two pieces of information. The first is a possible representation for the weights attached to each connection in the network, and the second is variable  $z$  to be added to the diagonal value of the within class scatter matrix to prevent it from being singular. In simple words, the connection weight matrix is represented by a linear genome formed by concatenating each of its rows. A population of 100 members was initially randomly generated. In order to bound the search space, the weight values were limited to a range between -1 and +1. This constraint also helps reduce the chance that the evolutionary process will produce a forced model with extreme weight values. The evolution process starts after initialization according to the DE equations mentioned above as shown in Figure.2 (A modified version of the one published by [7]). After computing the values of the connection weights for each node, the output of each node will be computed according to the following equation:

$$\mu_j(t) = f_t \sum_{i=0}^{n-1} w_{ij} x_i - \theta_j \quad (3)$$

In this equation,  $\mu_j(t)$  is the output of node  $j$  at time  $t$ ,  $x_i$  is the element  $i$  of the input, and  $f_t$  is the nonlinear transfer function chosen as the sigmoid function in this paper.  $\theta_j$  is the threshold value associated with each neuron, that can also be included in the genome linear representation.

One of the points that should be taken into consideration with feed-forward neural networks employing a sigmoid function is that care should be taken so that the maximum input to the nonlinear transfer function will not cause the output

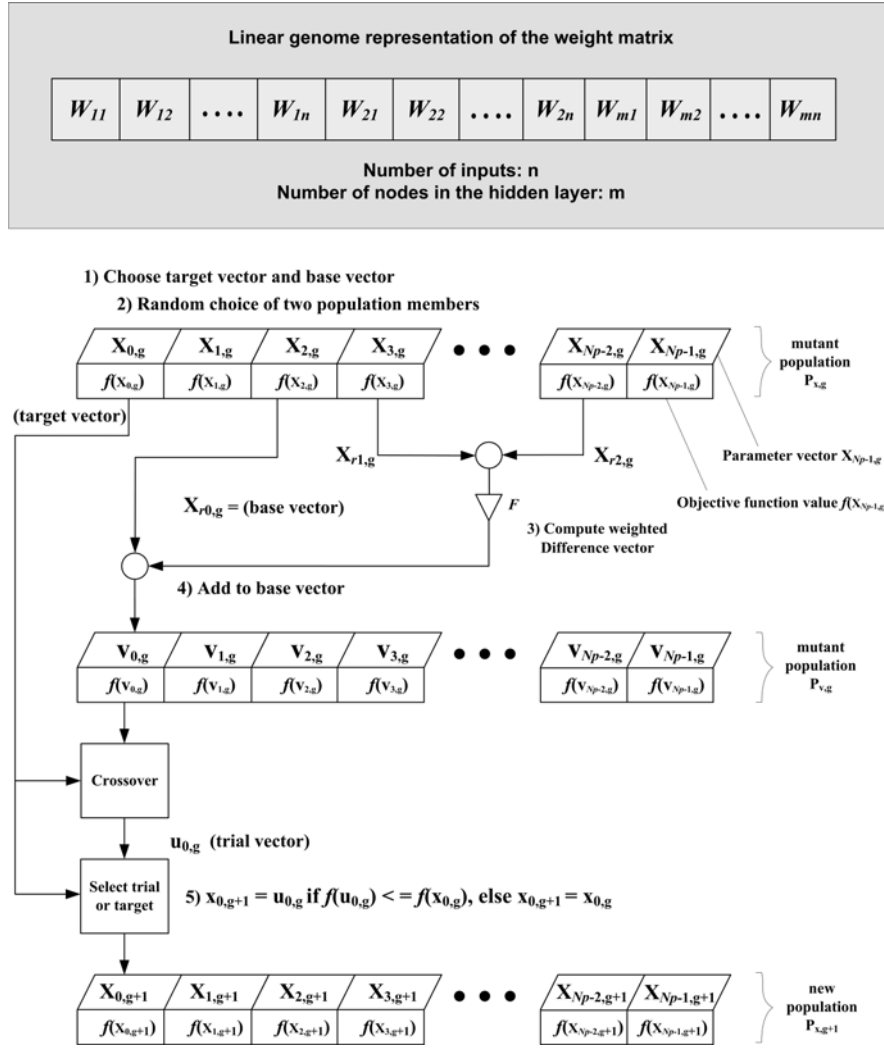


Fig. 2. DE based weight optimization technique

to saturate. Another point that affects the performance of the network is the number of nodes or neurons in the hidden layer. It is important to use enough neurons to capture the nonlinearities in the input, however, using too many neurons may cause an over fitting. In such a case the proposed neural network will not be able to generalize well on unseen data [13].

Since the weights of the proposed neural network that are evolved using the DE optimization technique require a fitness function to evaluate the importance of each member of the population, then the classification accuracy achieved by a suitable classifier is used here as a fitness function.

#### B. A New Fuzzy Linear Discriminant Analysis Projection Technique

Consider a classification problem with  $c$  classes, in which the data set of labelled training samples is given as:

$$S = \{(x_1, y_1), (x_2, y_2), \dots, (x_l, y_l)\} \subseteq (X, Y)^l \quad (4)$$

Where  $X$  is the input space and  $Y$  is the output space.  $X \subseteq \mathbb{R}^n$ , and  $l$  is the number of samples. Each training point  $x_i$ , where  $i = \{1, 2, 3, \dots, c\}$ , originally belongs to one of the  $c$  classes and is given a label  $y_i \in \{1, 2, 3, \dots, c\}$ . The goal is to find an optimal hyper-plane using the training samples that can recognize the test points, i.e., the classifier will have a good generalization capability. In FLDA each point,  $x_i$ , belongs to each of the  $c$  classes with a certain membership. The fuzzy within class scatter matrix  $S_W$ , fuzzy between class scatter matrix  $S_B$ , and the fuzzy total class scatter matrix  $S_T$  are given as follows [3]:

$$S_W = \sum_{i=1}^c \sum_{k=1}^{l_i} \mu_{ik}^m (x_k - v_i)(x_k - v_i)^T \quad (5)$$

$$S_B = \sum_{i=1}^c \sum_{k=1}^{l_i} \mu_{ik}^m (v_i - \bar{x})(v_i - \bar{x})^T \quad (6)$$

$$S_T = \sum_{i=1}^c \sum_{k=1}^{l_i} u_{ik}^m (x_k - \bar{x})(x_k - \bar{x})^T \quad (7)$$

where  $m$  (given that  $m > 1$ ) is the fuzzification parameter,  $u_{ik}$  is the membership of pattern  $k$  in class  $i$ ,  $x_{kj}$  is the value of the  $k$ 'th sample across the  $j$ 'th dimension,  $v_i$  is the mean of the patterns belonging to class  $i$ , and  $v_{ij}$  is its value across the  $j$ 'th dimension.  $\bar{x}$  is the mean of the training samples.

$$\bar{x} = \frac{1}{l} \sum_{k=1}^l x_k \quad (8)$$

In this paper, the value of the membership  $u_{ik}$  is calculated using a possibilistic fuzzy clustering approach. The cost function of the possibilistic clustering approach is adopted from [14], as given in Eq.(9) below.

$$J(\theta, U) = \sum_{k=1}^l \sum_{i=1}^c u_{ik}^m d(x_k, \theta_i) + \sum_{i=1}^c \eta_i \sum_{k=1}^l (1 - u_{ik})^m \quad (9)$$

where  $\theta_i$  is the  $i$ 'th cluster center,  $\eta_i$  is a positive constant that is suitably chosen for each class. The first term in Eq. (9) is the same objective function used in the probabilistic clustering approach, while the second term is added to reduce the effect of outliers. In order to find the membership values from the above equation, then the values of the clusters centers are needed. A direct way would be to differentiate Eq. (9) with respect to  $\theta_i$ , but this in turn would cancel the second term leaving only the first term. A general look at the first term of Eq. (9) reveals that it represents the classical within class scatter matrix  $S_W$  given in Eq. (5) if the weight is removed. Thus applying the values of the clusters means ensures that the objective function given by Eq. (9) would settle at a global optimum value. Then in order to compute the membership values, a differentiation of the resultant function with respect to  $u_{ik}$  needs to be done as follows.

$$\frac{\partial J(\theta, U)}{\partial u_{ik}} = m u_{ik}^{m-1} d(x_k, v_i) - m \eta_i (1 - u_{ik})^{m-1} = 0 \quad (10)$$

This would in turn result in the following function

$$u_{ik} = \frac{1}{1 + \frac{d(x_k, v_i)^{\frac{1}{m-1}}}{\eta_i}} \quad (11)$$

The values of  $\eta_i$ ,  $i = \{1, 2, 3, \dots, c\}$  were chosen to be equal to the maximum distance between the samples belonging to that class and the class center.

After computing all the variables, FLDA finds the vector  $G$  that would maximize the ratio of the between class scatter matrix to the within class scatter matrix by solving the following equation:

$$G = \arg \max_G \text{trace} \frac{G^T S_B G}{G^T S_W G}, \quad (12)$$

The solution can be readily computed by applying an Eigen-decomposition on  $S_W^{-1} S_B$ , provided that the within class scatter matrix  $S_W$  is nonsingular. In this paper, we are using a regularized version of  $S_W$  given by  $S_W = S_W + zI$ , for some  $z > 0$  that is included in the weight representation in Figure. 2, where  $I$  is an identity matrix. In this way the scatter

matrix is guaranteed to be nonsingular. Since the rank of the between class scatter matrix is bounded from above by  $c - 1$ , there are at most  $c - 1$  discriminant vectors by FLDA.

### III. EXPERIMENTS AND PRACTICAL RESULTS

The experiments section is decomposed mainly into two parts. In the experiments, different datasets and classifiers are used to prove the effectiveness of the proposed nonlinear fuzzy discriminant analysis presented in this paper that will be referred to as NFDA in the experiments. The details of the experiments carried on are listed below:

- Comparison with other methods: The performance of the proposed NFDA will be compared against different dimensionality reduction techniques from the literature. Due to the wide variety of dimensionality reduction methods in the literature, we will only select some of these methods. The chosen methods from literature were: Kernel Discriminant Analysis (KDA) [15], Kernel Principal Components Analysis (KPCA)[16], Orthogonal Linear Discriminant Analysis (OLDA) [17], Uncorrelated Linear Discriminant Analysis (ULDA) [18], and Fuzzy Linear Discriminant Analysis (FLDA) [3].
- Datasets employed: Since this work aims to present a novel variation to the existing techniques, then a comparison with the existing techniques is necessary on different datasets before employing it on the prosthetic control problem. The following datasets are utilized for this purpose, these are:
  - Group-1: These are acquired from the UCI Machine Learning Repository ([www.ics.uci.edu/~mllearn/mlrepository.html](http://www.ics.uci.edu/~mllearn/mlrepository.html)) with different number of samples and numbers of features. The type of the classifier chosen for these datasets is a K-Nearest Neighbor classifier (KNN), with  $k=5$ .
  - Group-2: Electroencephalogram (EEG) datasets measured specifically for the purpose of this research. Details will be given in the appropriate section later. The type of the classifier chosen with these datasets is a linear discriminant analysis (LDA) classifier.
- Testing method employed: The general testing scheme employed is a three way data split. The dataset utilized is divided into three sets: training, validation, and testing. An initial projection matrix is calculated based on the training set. Then a validation set is used in order to optimize the weights to produce the optimum projection matrix that can minimize the mean of the training and validation errors. Finally a completely unseen testing set is utilized to measure the generalization capability of the proposed system.

#### A. Experiments on UCI Datasets

Each dataset taken from the UCI Repository is subdivided into three parts, with the percentage of the data forming each of the training, validation and testing given as 20%,

20% and 60% respectively. The results shown in Table.I and Table.II were acquired by utilizing a K-Nearest neighbor (KNN) classifier with the number of neighbors being 5. In order to interpret the results, one can start by analyzing the performance of ULDA and OLDA. The performance of these two methods was found to be the same for the different datasets, and very close to that of FLDA. This is justified by the fact that both are implemented in the nearly same way with the difference being that OLDA applies a QR decomposition as a final step [17]. On the other hand, the performance of the kernel approaches, KDA and KPCA, was in general better than FLDA, ULDA, and OLDA due to the use of the kernel tricks. In comparison, NFDA managed to achieve the highest performance for most of the datasets and it was slightly behind the KDA for the remaining few ones. However, unlike KDA the proposed NFDA method does not require the kernel matrices, which makes it computationally more efficient.

TABLE I  
VALIDATION SET ERROR RESULTS ON DATA OBTAINED FROM THE UCI  
REPOSITORY AVERAGED ACROSS 10 RUNS

Dataset	FLDA	ULDA	OLDA	NFDA	KDA	KPCA
Pendigits	16.47	16.18	16.18	2.12	2.06	15.27
Magic	20.53	20.55	20.55	14.61	14.41	21.35
Terma	0.00	0.00	0.00	0.00	13.70	0.00
German	25.87	25.87	25.87	13.43	29.35	26.87
Cancer	2.17	2.17	2.17	1.45	6.52	4.35
Dermatology	2.70	2.70	2.70	0.00	9.46	1.35
Hill_Valley	36.21	36.21	36.21	7.00	24.69	30.04

TABLE II  
TESTING SET ERROR RESULTS ON DATA OBTAINED FROM THE UCI  
REPOSITORY AVERAGED ACROSS 10 RUNS

Dataset	FLDA	ULDA	OLDA	NFDA	KDA	KPCA
Pendigits	17.78	18.07	18.07	3.37	2.12	16.47
Magic	19.59	19.59	19.59	13.55	13.25	19.62
Terma	5.71	5.71	5.71	2.86	2.86	4.29
German	27.14	28.14	28.14	27.64	29.15	33.67
Cancer	4.44	4.44	4.44	2.22	6.67	2.96
Dermatology	2.78	4.17	4.17	2.78	5.56	4.17
Hill_Valley	28.93	28.93	28.93	6.61	21.90	26.03

### B. A Neurobotic Controller Employing NFDA

The emerging field of *neurorobotics* seeks to obtain motor command signals from motor control regions of the brain and transform them into electronic signals suitable for controlling a robotic device [19]. The primary motor cortex (MI), in the precentral gyrus of the human cerebral cortex, has long been known to be important for the control of voluntary limb movements. It is therefore conceivable that one could record commands for arm movement in the MI cortex and use those signals to directly drive a robotic arm of similar configuration. Specifically, the Electroencephalogram (EEG) signal measured from electrodes placed on the human scalp is considered for this research.

The field of neuroprosthetics has grown rapidly to include a variety of assistive and rehabilitation devices designed to help the disabled people or the amputees with their daily tasks. Brain Machine Interface (BMI) requires effective processing

of the EEG measurements. The most widely used approach to interpret the EEG signals employs a pattern recognition scheme as shown in Figure.3. In such a scheme there are five main tasks to be performed, these are: pre-processing, feature extraction, dimensionality reduction, pattern classification, and finally mapping the classifier output to a suitable robot arm command. The combination of ANN and FLDA mentioned in the earlier sections fits within the third task, as it aims to produce a low dimensional feature set with better discrimination power.

The EEG dataset was recorded using two EEG channels and processed by the ProComp2 encoder from Thought Technology Ltd. The system was chosen because we believe that it would be much more practical in terms of everyday setup for a real-life user. Another reason is that we wish to limit the data supplied to the system in real time so that we can process it easily on an embedded platform.

Five subjects participated in the experiments, with approval from the University Human Research Ethics Committee and informed consent from the volunteers. Electrodes were placed on the C3 and C4 locations that are known from the literature to show the most prominent changes for motor imagery data. Each user was instructed to imagine three different classes of the arm movement, these are: Elbow Flexion, Pen Grip, and Hand Open as shown in Figure 4. The users were asked to perform around 12 trials of imagining each of these classes. Within each trial, a total of 30 seconds of data were recorded at 256 Hz sampling rate.

In the feature extraction stage, different features were extracted from EEG signals in the literature. Some of these methods are: Power estimate [20], Autoregressive model features (AR) [21], Wavelet Transform (WT) and the Wavelet Packet Transform (WPT) based feature set [22]. In this paper, a windowing scheme was adopted in which a sliding window was incremented each time and features were extracted from each window. Different window lengths (128, 256, and 384 samples) were adopted to test the effectiveness of the proposed technique under various situations. These windows were incremented by 64 samples each time. The extracted feature set included a combination of AR features with additional time domain features like skewness (SKEW), mean average value (MAV), waveform length (WL), and root mean square (RMS). The reason for selecting such a combination of features is that it does not need large computational power like the WT and WPT feature, while at the same time being an effective feature set. The total number of extracted features were 10 from each channel, thus 20 features were extracted from the two channels (10 features/channel = 6 AR + SKEW + MAV + WL + RMS).

In the dimensionality reduction stage, different techniques were employed to present a fair comparison. These included: Kernel Discriminant Analysis (KDA)[15], Kernel Principal Components Analysis (KPCA)[16], Neighborhood Preserving Embedding (NPE)[23], and Kernel Locality Preserving Projection (KLPP) [24]. Also included was the MLP trained with back propagation algorithm. The MLP was added as it employs a nonlinear mapping internally within its hidden layers, thus a comparison with MLP was necessary. All of these methods were compared with the one proposed in this

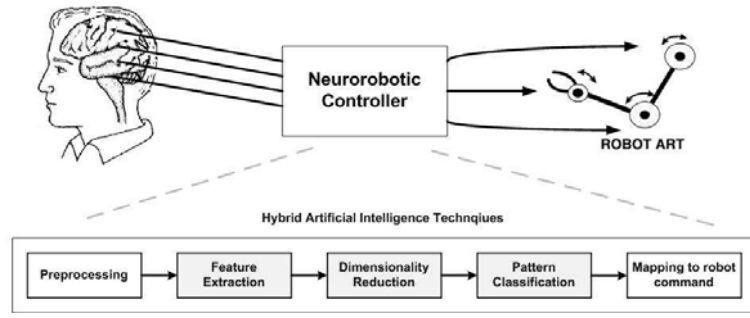


Fig. 3. Block diagram of a neurorobotic control system

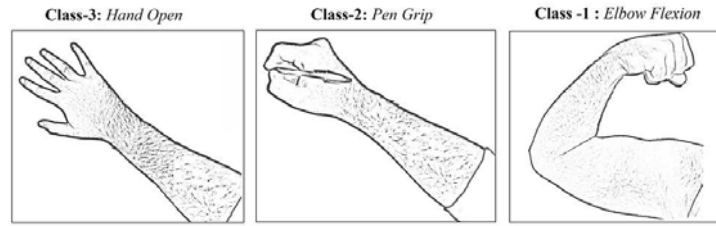


Fig. 4. Different classes of hand movements that the user imagined during the experiments.

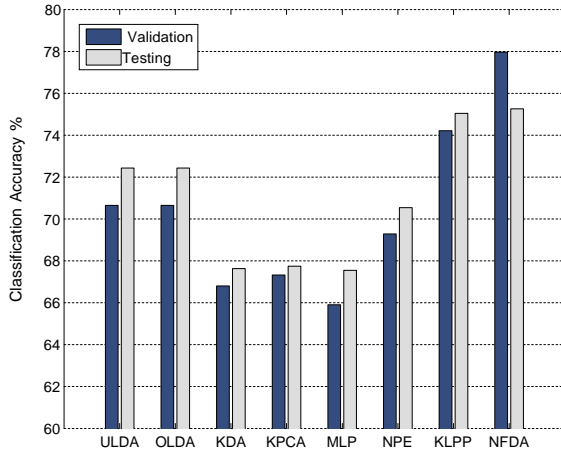
paper, referred to as NFDA. The testing scheme employed included a three way data split in which the total data was divided into training, validation, and testing. The objective function was to minimize both the training and validation errors and the difference between them. Then the network was tested with the completely unseen testing set to measure the generalization capability of the system. An important note to mention here is the number of neurons utilized within the hidden layer, which was roughly set to three times the number of features, as this proved to present powerful results.

In the first part of this experiment, for each subject the training and validation sets were made equal to the first 40% of the total data only, i.e., 20% for the training set and 20% for the validation set. The rest of the data comprising 60% of the total extracted feature set was assigned to the testing set. This was done in order to check the generalization capability of the system when trained with small data size. In this case, the average classification accuracy results across five subjects are shown in Fig.5. These results were computed for three different analysis windows lengths comprising 128 samples (i.e.,  $128/256 \text{ Hz} = 0.5 \text{ sec}$ ), 256 samples (i.e.,  $256/256 \text{ Hz} = 1 \text{ sec}$ ), and 384 samples (i.e.,  $384/256 \text{ Hz} = 1.5 \text{ sec}$ ) each increased with 64 samples (i.e.,  $64/256 = 0.25 \text{ sec}$ ). In order to analyze the results, one can start by looking at the effect of the windows length on the classification accuracies achieved by all methods. When considering a window length of 128 sample, the performance of each of the KDA, KPCA, and MLP is shown to be worse than that of the NPE, ULDA, and OLDA. These methods were in turn outperformed by KLPP which is in turn outperformed by NFDA. This is justified by the fact that the performance of each of the KDA, PCA, and MLP is sensitive to the analysis windows length and that these methods usually requires fairly large amount of training data

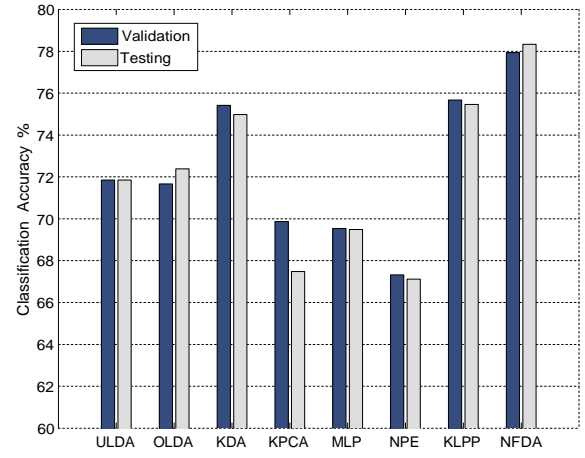
in order to generalize well on unseen testing data, while this wasn't the case with the current experiment. Additionally, it is also known that the MLP cannot escape a local minima if it encounters one during its iterative training procedure. On the other hand, KLPP is shown to outperform these methods and this may be justified by its ability to preserve the local structure of the data points in addition to the use of the kernel trick. When increasing the analysis windows length to 256 or 384 samples, the performance of KDA is clearly enhanced (showing better performance than NPE, KPCA, and MLP) and is capable to compete with ULDA and OLDA, while all of these methods (i.e., KDA, OLDA, and ULDA) are still performing worse than KLPP and NFDA. In comparison, the performance of NFDA was optimized with DE to find the nonlinear mapping that can well separate the problem classes. Thus, NFDA was able to provide a better separation between the data points from different classes across different windows lengths.

In the next part of this experiment, the data divisions size for the training, validation, and testing were varied. The training set was made to be 60% of the total data, and the rest 40% was equally divided between the validation set 20% and the testing set 20%. This was made in order to have a better look at the performance of methods like KDA that is dependent on the training data size. The average classification accuracies across five subjects are shown in Fig.6.

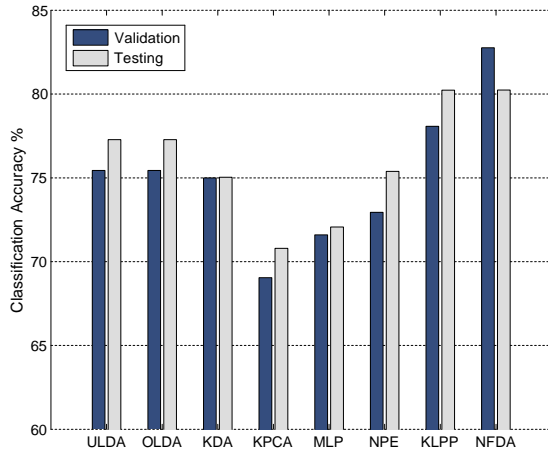
It is very clear that the performance of all the methods was enhanced, due to the increase of the training set size, and this is especially obvious for KDA. In comparison the proposed NFDA was again capable of maintaining its superior performance even in this case giving a testing accuracy of 78.33%, 82.97%, and 85.77% for a window length of 128, 256, and 384 samples respectively. Again this is due to the



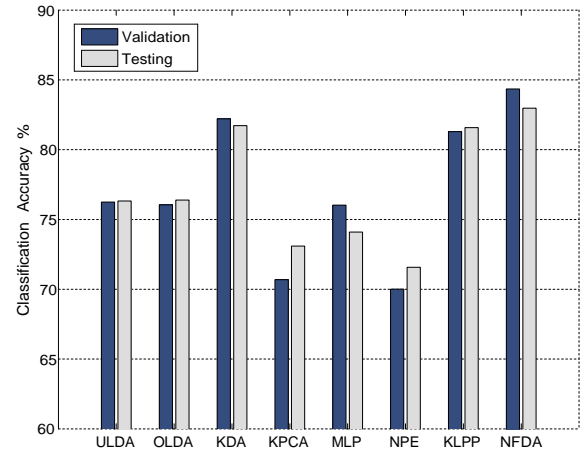
(a) Window Length =128 msec



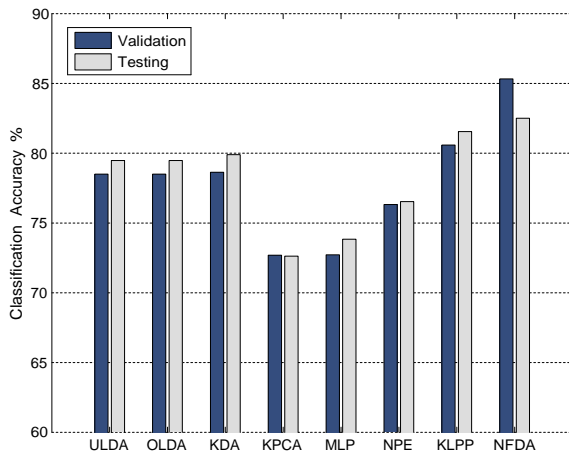
(a) Window Length =128 msec



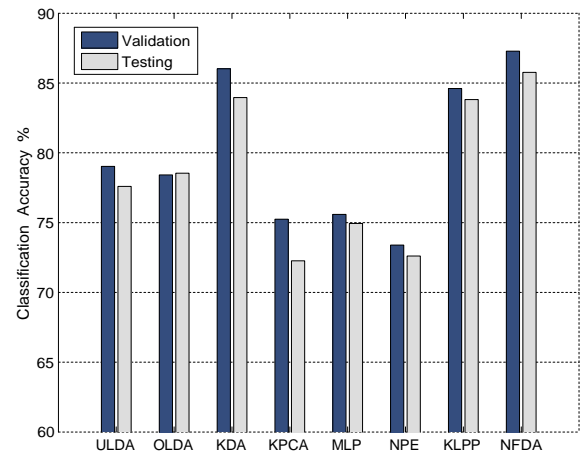
(b) Window Length =256 msec



(b) Window Length =256 msec



(c) Window Length =384 msec



(c) Window Length =384 msec

Fig. 5. Classification accuracies averaged across 5 subjects with 20%, 20%, 60% divisions

Fig. 6. Classification accuracies averaged across 5 subjects with 60%, 20%, 20% divisions

use of a DE-based nonlinear layer that made it possible for NFDA to achieve the best results.

Thus, all of the reported results proves the capability of the proposed hybrid NFDA when dealing with different window lengths, and also its effectiveness on different datasets.

#### IV. CONCLUSION

In this paper, a new nonlinear discriminant analysis based feature projection technique was proposed. The new technique included a hybrid of neural networks and Fisher's discriminant analysis. The theory and justification behind this technique was explained. The algorithm was compared with other statistical techniques and multilayer perceptron, on a number of benchmark datasets and a brain machine interface problem with three classes of imagination. On average, the proposed technique managed to achieve better results than all other methods even the kernel based discriminant analysis (82.50% for NFDA, 79.89% for KDA). The results indicate that by properly training a single layer neural network and mixing it with FLDA, a powerful combination can be achieved for feature projection purposes. More experiments will be conducted in the future as we are currently extending this technique to have a self tuning capability.

#### REFERENCES

- [1] S. Theodoridis and K. Koutroumbas, *Pattern Recognition*, 3rd ed. Academic Press, 2006.
- [2] J. Watada, H. Tanaka, and K. Asai, *Fuzzy discriminant analysis in fuzzy groups*, *Fuzzy Sets and Systems*, Vol. 19, No. 3, pp.261–271, 1986.
- [3] H. X. Wua and J. J. Zhoua, *Fuzzy discriminant analysis with kernel methods*, *Pattern Recognition*, Vol. 39, No. 11, pp.22362239, 2006.
- [4] J. W. Lu, K. N. Plataniotis, and A. N. Venetsanopoulos, *Face recognition using kernel direct discriminant analysis algorithms*, *IEEE Transactions on Neural Networks*, Vol. 14, No. 1, pp. 117–126, 2003.
- [5] T. Xiong, J. Ye, and V. Cherkassky, *Kernel uncorrelated and orthogonal discriminant analysis: A unified approach*, *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 125-131, 2006.
- [6] Z. Liang and P. Shi, *Uncorrelated discriminant vectors using kernel method*, *Pattern Recognition*, Vol. 38, No. 2, pp. 307–310, 2005.
- [7] K. V. Price, R. M. Storn, and J. A. Lampinen, *Differential evolution: A practical approach to global optimization*, Springer, 2005.
- [8] A. R. Webb and D. Lowe, *The optimised internal representation of multilayer classifier networks performs nonlinear discriminant analysis*, *Neural Networks*, Vol. 3, No. 4, pp. 367-375, 1990.
- [9] P. Gallinari, S. Thiria, F. Badran, and F. Fogelman-Foulie, *On the relations between discriminant analysis multilayer perceptrons*, *Neural Networks*, Vol. 4, No. 3, pp. 349–360, 1991.
- [10] D. Casasent, and X. Chen, *Radial basis function neural networks for nonlinear fisher discrimination and neyman-pearson classification*, *Neural Networks*, Vol. 16, Volume 16, No. 5-6, pp. 529-535, 2003.
- [11] Y. Kwon, and B. Moon, *Nonlinear feature extraction using a neuro genetic hybrid*, *Proceedings of the 2005 conference on Genetic and evolutionary computation*, Washington DC, USA, pp. 2089–2096, 2005.
- [12] A. K. Palit and D. Popovic, "Computational intelligence in time series forecasting: theory and engineering applications", Springer, 2005.
- [13] L. H. Tsoukala, and R. E. Uhrig, *Fuzzy and Neural Approaches in Engineering (Adaptive and Learning Systems for Signal Processing, Communications and Control Series)*, John Wiley and Sons, 1997.
- [14] J. V. D. Oliveira, and W. Pedrycz, *A comprehensive, coherent, and in depth presentation of the state of the art in fuzzy clustering*, John Wiley and Sons Ltd, 2007.
- [15] G. Baudat, and F. Anouar, *Generalized discriminant analysis using a kernel approach*, *Neural Computation*, Vol. 12, No. 10, pp. 2385–2404, 2000.
- [16] B. Scholkopf, A. Smola, K.-L. Miller, *Nonlinear Component Analysis as a Kernel Eigenvalue Problem*, *Neural Computation*, Vol. 10, pp. 1299-1319, 1998.
- [17] J. Ye and T. Xiong, *Computational and theoretical analysis of null space and orthogonal linear discriminant analysis*, *Journal of Machine Learning Research*, Vol. 7, pp. 1183-1204, 2006.
- [18] J. Ye, R. Janardan, Q. Li, and H. Park, *Feature Extraction via Generalized Uncorrelated Linear Discriminant Analysis*,
- [19] J. Chapin, and K. Moxon, *Neural prostheses for restoration of sensory and motor function*, CRC Press LLC, 2001.
- [20] G. Pfurtscheller, C. Neuper, and D. Flotzinger, *EEG-based discrimination between imagination of right and left hand movement*, *Electroencephalography and Clinical Neurophysiology*, Vol. 103, No. 6, pp. 642–651, 1997.
- [21] G. S. Dharwarkar, and O. Basir, *Enhancing temporal classification of AAR parameters in EEG single-trial analysis for brain-computer interfacing*, *Proceedings of The 28th IEEE EMBS Annual International Conference*, New York City, USA, pp. 5358–5361, 2005.
- [22] R. N. Khushaba, A. Al-Sukker, A. Al-Ani, A. Al-Jumaily, *Intelligent artificial ants based feature extraction from wavelet packet coefficients for biomedical signal classification*, *3rd International Symposium on Communications, Control and Signal Processing, ISCCSP*, Malta, pp. 1366–1371, 2008.
- [23] H. Xiaofoei, C. Deng, Y. Shuicheng, and H. J. Zhang, *Neighborhood Preserving Embedding*, *Tenth IEEE International Conference on Computer Vision (ICCV'2005)*, pp. 1208-1213, 2005.
- [24] X. He, *Locality Preserving Projections*, PhD thesis, Computer Science Department, The University of Chicago, 2005.